HAL open science

Bidimensionality and Parameterized Algorithms

Dimitrios M. Thilikos

▶ To cite this version:

Dimitrios M. Thilikos. Bidimensionality and Parameterized Algorithms. IPEC 2015 - 10th International Symposium on Parameterized and Exact Computation, Sep 2015, Patras, Greece. pp.1-16, 10.4230/LIPIcs.IPEC.2015.1 . lirmm-01370293

HAL Id: lirmm-01370293

https://hal-lirmm.ccsd.cnrs.fr/lirmm-01370293

Submitted on 22 Sep 2016

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Distributed under a Creative Commons Attribution 4.0 International License

Bidimensionality and Parameterized Algorithms*

Dimitrios M. Thilikos[1,2,3]

1 **AlGCo Project Team, CNRS, LIRMM, Montpellier, France**
sedthilk@thilikos.info
2 **Department of Mathematics, University of Athens, Athens, Greece**
3 **Computer Technology Institute & Press "Diophantus", Patras, Greece**

Abstract

We provide an exposition of the main results of the theory of bidimensionality in parameterized algorithm design. This theory applies to graph problems that are bidimensional in the sense that i) their solution value is not increasing when we take minors or contractions of the input graph and ii) their solution value for the (triangulated) $(k \times k)$-grid graph grows as a quadratic function of k. Under certain additional conditions, mainly of logical and combinatorial nature, such problems admit subexponential parameterized algorithms and linear kernels when their inputs are restricted to certain topologically defined graph classes. We provide all formal definitions and concepts in order to present these results in a rigorous way and in their latest update.

1998 ACM Subject Classification G.2.1 Combinatorics, G.2.2 Graph Theory

Keywords and phrases Parameterized algorithms, Subexponential FPT-algorithms, Kernelization, Linear kenrels, Bidimensionality, Graph Minors

Digital Object Identifier 10.4230/LIPIcs.IPEC.2015.1

Category Invited Paper

1 Introduction

The theory of bidimensionality, was introduced in [27] and has been developed further during the last decade in [33, 35, 28, 38, 55, 59, 67, 54, 53] (see also [30, 34, 52, 26, 39, 32]). It provides general techniques for designing efficient fixed-parameter algorithms and approximation schemes for NP-hard graph problems in broad classes of graphs.

A parameterized problem on graphs can be seen as a subset Π of $\mathcal{G} \times \mathbb{N}$ where $\mathcal{G}$ is some graph class (for instance, planar graphs) and the question is whether an instance (G, k) is a member of Π, where k is the parameter of the problem. The main objective is to design an $f(k) \cdot n^{O(1)}$-step algorithm that answers this question while keeping the parametric dependence $f(k)$ as low as possible. This implies that, for each fixed value k, the problem can be solved by an algorithm running in polynomial-time where the degree of this polynomial does not depend on the value of k.

The combinatorial base of bidimensionality is the celebrated grid-exclusion theorem from the Graph Minors series of Robertson and Seymour [82, 81]. This theorem states that every graph excluding a $(r \times r)$-grid as a minor should have treewidth bounded by some function of r (see Subsection 2.1 for the formal definition of treewidth and the minor relation). Treewidth

* The work of this paper was co-financed by the European Union (European Social Fund ESF) and Greek national funds through the Operational Program "Education and Lifelong Learning" of the National Strategic Reference Framework (NSRF) - Research Funding Program: ARISTEIA II.

© Dimitrios M. Thilikos (Δημήτριος Μ. Θηλυκός);
licensed under Creative Commons License CC-BY
10th International Symposium on Parameterized and Exact Computation (IPEC 2015).
Editors: Thore Husfeldt and Iyad Kanj; pp. 1–16
Leibniz International Proceedings in Informatics
LIPICS Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

is a cornerstone parameter in algorithmic graph theory measuring the topological resemblance of a graph to the structure of a tree.

The central idea of bidimensionality resides in the fact, that for many parameterized graph problems, the presence in their input graphs of a (bidimensional) $(\Omega(\sqrt{k}) \times \Omega(\sqrt{k}))$-grid as a minor is directly providing a positive (or a negative) answer to the problem. The bidimensionality condition, together with certain conditions on $\mathcal{G}$, is able to reduce the problem to the case where the treewidth of the input graph is sublinear in the problem parameter k.

A graph of bounded treewidth can be viewed as a "monodimensional" tree-like structure. According to Courcelle's theorem [21], if the problem is expressible in Monadic Second Order Logic (MSO), then it is possible to process this tree-like structure as the input of a tree automaton that can solve the problem in time that is linear in the size of the input graph. If the parametric dependence of this algorithm can be made singly exponential, the sublinear (on k) treewidth of G yields a parameterized algorithm with subexponential parameterized dependence. This simple reasoning, provides a generic way to design parameterized algorithms with subexponential parametric dependence. In many cases, this provides algorithms running in $2^{O(\sqrt{k})} \cdot n^{O(1)}$ which appears to be the best parametric dependence one may expect, according to the results in [14].

In Section 2 we provide all definitions and theorems that support the above ideas. The concept of bidimensionality is formally defined in Section 3 and in Section 4 we abstract the above methodology into a single theorem on subexponential parameterized algorithms (Theorem 7).

Another, somehow more technical, application of bidimensionality is *kernelization*. A kernelization algorithm for a parameterized graph problem Π is a polynomial-time algorithm that reduces every instance (G, k) to an equivalent one (a kernel) whose size is bounded only by a function of k. When this function is linear on k, we say that Π admits a linear kernel (see Subsection 5 for the formal definitions). Kernelization has been a vibrant field of parameterized complexity and a lot of research has been oriented to the derivation of linear kernels for parameterized problems. Bidimensionality theory has meta-algorithmic applications in the derivation of linear kernels. It follows that, given a parameterized problem $\Pi \subseteq \mathcal{G} \times \mathbb{N}$ where $\mathcal{G}$ satisfies certain (topological) conditions, a linear kernel is automatically derived when Π is bidimensional, is expressible in Counting Monadic Second Order Logic, and satisfies some separability condition. We describe this result in Section 5. For this, we present the basic tools supporting it, namely, the notions of protrusion decomposition and protrusion replacement. We also point out some methodological analogies with the previous case of subexponential algorithms, mainly in what concerns the classification of the required tools into algorithmic and combinatorial ones.

In our exposition we present the contributions of bidimensionality theory to parameterized algorithms in their most general, up to now, version. In contrast to previous surveys on this topic [34, 52], we preferred to insist on the rigorous mathematical formalization of this theory which may require (not only for the unexperienced reader) to go through the definitions of Sections 3 and 4. The exposition concludes by some open problems and further directions in Section 6.

2 Basic concepts

In this section we give some basic definitions that are necessary for the exposition of the rest of the paper.

2.1 Graphs

All graphs in this paper are undirected and without multiple edges or loops. Given a graph G, we use the notation $V(G)$ and $E(G)$ for the vertex set and the edge set of G respectively. We say that a graph H is a *subgraph* of G if $V(H) \subseteq V(G)$ and $E(H) \subseteq E(G)$. Given a set $S \subseteq V(G)$ we denote by $G[S]$ the subgraph G' of G where $V(G') = S$ and $E(G') = \{\{x, y\} \in E(G) \mid \{x, y\} \subseteq S\}$ and we call G' the *subgraph of G induced by S* or we simply say that G' is an induced subgraph of G. Given a set $S \subseteq V(G)$, we denote by $\partial_G(S)$ the set of all vertices in S that are adjacent in G with vertices not in S. We also define the neighborhood of S in G by $N_G(S) = \partial_G(V(G) \setminus S)$.

Treewidth. A *tree decomposition* of a graph G is a pair $\mathcal{T} = (T, \{X_t\}_{t \in V(T)})$, where T is a tree whose every node t is assigned a vertex subset $X_t \subseteq V(G)$, called a bag, such that the following three conditions hold:

- $\bigcup_{t \in V(T)} X_t = V(G)$, i.e., every vertex of G is in at least one bag.
- For every $\{u, v\} \in E(G)$, there exists a node t of T such that $u, v \in X_t$.
- For every $u \in V(G)$, the graph $T[\{t \in V(T) \ : \ u \in X_t\}]$ is connected.

The *width* of a tree decomposition $\mathcal{T} = (T, \{X_t\}_{t \in V(T)})$ equals $\max_{t \in V(T)} |X_t| - 1$. The *treewidth* of a graph G, denoted by $\mathbf{tw}(G)$, is the minimum possible width of a tree decomposition of G.

Minors and contractions. Given an edge $e = \{x, y\}$ of a graph G, the graph G/e is obtained from G by contracting the edge e, that is, the endpoints x and y are replaced by a new vertex $v_{x,y}$ which is adjacent to the old neighbors of x and y (except from x and y). A graph H obtained by a sequence of edge-contractions is said to be a *contraction* of G. We denote it by $H \leq_c G$. A graph H is a *minor* of a graph G if H is the contraction of some subgraph of G and we denote it by $H \leq_m G$. We say that a graph G is *H-minor-free* when it does not contain H as a minor. We also say that a graph class $\mathcal{G}$ is *H-minor-free* (or, excludes H as a minor) when all its members are H-minor-free. A graph G is an *apex graph* if there exists a vertex v such that $G \setminus v$ is planar. A graph class $\mathcal{G}$ is *apex-minor-free* if there exists an apex graph H that is not in $\mathcal{G}$.

Grids and triangulated grids. Given a positive integer k, we denote by $\boxplus_k$ the $(k \times k)$-grid. Formally, for a positive integer k, a *$(k \times k)$-grid* $\boxplus_k$ is a graph with vertex set $\{(x, y) \ : \ x, y \in \{1, \ldots, k\}\}$. Thus $\boxplus_k$ has exactly k^2 vertices. Two different vertices (x, y) and (x', y') are adjacent if and only if $|x - x'| + |y - y'| = 1$.

For an integer $t > 0$, the graph Γ_t is obtained from the grid $\boxplus_t$ by adding, for all $1 \leq x, y \leq t - 1$, the edge $(x+1, y), (x, y+1)$, and additionally making vertex (t, t) adjacent to all the other vertices (x, y) with $x \in \{1, t\}$ or $y \in \{1, t\}$, i.e., to the whole border of $\boxplus_t$. Graph Γ_9 is shown in Fig. 1.

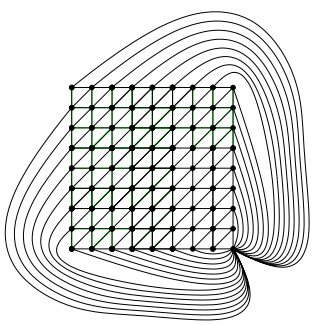

Figure 1 The graph Γ_9.

2.2 Properties of graph classes

A graph class $\mathcal{G}$ is said to be *minor-closed/contraction-closed* if every minor/contraction of a graph in $\mathcal{G}$ also belongs to $\mathcal{G}$.

In general, it is known that there exists a constant c such that any graph G which excludes a $\boxplus_t$ as a minor has treewidth at most $O(t^c)$. The exact value of c remains unknown, but it is more than 2 and at most 36 [15, 20], while it is believed that $c \leq 3$ [36]. We will restrict our attention to graph classes on which $c < 2$ as it is then when bidimensionality theory applies. In particular we say that a graph class $\mathcal{G}$ has the *subquadratic grid minor property* (**SQGM** property for short) if there exist constants $\lambda > 0$ and $1 \leq c < 2$ such that any graph $G \in \mathcal{G}$ which excludes $\boxplus_t$ as a minor has treewidth at most λt^c.

Problems that are contraction-closed but not minor-closed are considered on more restricted classes of graphs. We say that a graph class $\mathcal{G}$ has the *subquadratic gamma contraction* (**SQGC** property for short) if there exist constants $\lambda > 0$ and $1 \leq c < 2$ such that any *connected* graph $G \in \mathcal{G}$ excluding Γ_t as a contraction has treewidth at most λt^c.

The following proposition, for the case of **SQGM**, follows directly from the linearity of excluded grid-minor in H-minor-free graphs proven by Demaine and Hajiaghayi [35], while for the case if **SQGC** it follows from [54].

▶ **Proposition 1.** *For every graph H, H-minor-free graph class $\mathcal{G}$ has the **SQGM** property for some λ depending on H and with $c = 1$. If H is an apex graph, then $\mathcal{G}$ has the **SQGC** property for some λ depending on H and with $c = 1$.*

Notice hat every graph class G with the **SQGC** property has the **SQGM** property. Clearly, the class of planar graphs has both above properties as there is an apex graph containing both K_5 and $K_{3,3}$ as a minor.

Recently, graph classes with the **SQGM** property that are not defined in the context of minor exclusion where detected. In [57] it was proven that unit disk graphs with maximum degree Δ have the **SQGM** property for some λ depending on Δ and with $c = 1/2$. This result has been extended for more general families of geometric intersection graphs in [67].

2.3 Parameterized problems on graphs

Parameterized problems. A parameterized problem Π can be seen as a subset of $\Sigma^* \times \mathbb{N}$ (we denote by $\mathbb{N}$ the set of all non-negative integers). We say that two instances (x, k) and (x', k') of some parameterized problem Π are *equivalent* if and only if $(x, k) \in \Pi \iff (x', k') \in \Pi$.

Parameterized tractable problems. Let Π be a parameterized problem. We say that Π is *fixed parameter tractable* if there exists a function $f : \mathbb{N} \to \mathbb{N}$ and an algorithm deciding whether $(x, k) \in \Pi$ (i.e., whether (x, k) is a YES-instance of Π) in $f(k) \cdot |x|^{O(1)}$ steps. We call such an algorithm FPT-*algorithm*. A parameterized problem belongs to the parameterized class FPT if it can be solved by an FPT-algorithm. (See the monographs [46, 77, 50, 23] on parameterized algorithms and complexity.)

Parameterized graph problems. We say that a parameterized problem Π is a *parameterized graph problem* when in each instance $(x, k) \in \Pi$, x encodes a graph. From now on, we deal with parameterized graph problems as subsets of $\mathcal{G}_{\text{all}} \times \mathbb{N}$ where $\mathcal{G}_{\text{all}}$ is the set of all graphs. Let $\mathcal{G}$ be a class of graphs, i.e, $\mathcal{G} \subseteq \mathcal{G}_{\text{all}}$. The *restriction* of a parameterized problem Π to $\mathcal{G}$ is defined as $\Pi \Cap \mathcal{G} = \{(G, k) \mid (G, k) \in \Pi \text{ and } G \in \mathcal{G}\}$.

2.4 Counting Monadic Second Order Logic

The syntax of Monadic Second Order Logic (MSO) of graphs includes the logical connectives $\vee$, $\wedge$, $\neg$, $\Leftrightarrow$, $\Rightarrow$, variables for vertices, edges, sets of vertices, and sets of edges, the quantifiers $\forall$, $\exists$ that can be applied to these variables, and the following five binary relations:

1. $u \in U$ where u is a vertex variable and U is a vertex set variable;
2. $d \in D$ where d is an edge variable and D is an edge set variable;
3. $\mathbf{inc}(d, u)$, where d is an edge variable, u is a vertex variable, and the interpretation is that the edge d is incident with the vertex u;
4. $\mathbf{adj}(u, v)$, where u and v are vertex variables and the interpretation is that u and v are adjacent;
5. equality of variables representing vertices, edges, sets of vertices, and sets of edges.

In addition to the usual features of monadic second-order logic, if we have atomic formulas testing whether the cardinality of a set is equal to q modulo r, where q and r are integers such that $0 \leq q < r$ and $r \geq 2$, then this extension of the MSO is called *counting monadic second-order logic*. Thus CMSO is MSO enriched with the following atomic formula for a set S: $\mathbf{card}_{q,r}(S) = \mathbf{true}$ if and only if $|S| \equiv q \pmod{r}$. We refer to [4, 21, 22] for a detailed introduction on CMSO and its algorithmic consequences.

3 Bidimensionality

In this section we define all concepts that are necessary for the definition of the bidimensionality property of parameterized graph problems.

3.1 Subset problems

A *vertex subset* (resp. *edge subset*) *certifying function* ϕ is a computable function which takes as input a graph G and a set $S \subseteq V(G)$ (resp. a set $S \subseteq E(G)$) and outputs true or false.

A *vertex* (resp. *edge*) *subset minimization/maximization problem* Π is a parameterized problem on graphs for which there exists a vertex (resp. edge) certifying function ϕ such that for every $(G, k) \in \mathcal{G} \times \mathbb{N}$ it holds that $(G, k) \in \Pi$ if and only if there exists a set $S \subseteq V(G)$ (resp. $S \subseteq E(G)$) such that $|S| \leq k$ for minimization problems (or $|S| \geq k$ for maximization problems) so that $\phi(G, S) = \mathsf{true}$. If, additionally, there exists a CMSO formula ψ such that $\phi(G, S) = \mathsf{true}$ if and only if $(G, S) \models \psi$, then we say that Π is a MIN-CMSO problem (or a MAX-CMSO problem).

For an example, for the Dominating Set problem we have that $\phi(G, S) = \mathsf{true}$ if and only if $\forall v \in V(G)(v \in S \vee \exists u \in V(G) : \mathbf{adj}(v, u))$. Therefore Dominating Set is a vertex subset minimization problem that is also as min-CMSO problem.

For simplicity, we will also use the term *subset problems* instead of vertex or edge subset minimization/optimization problems. Let us remark that there are many subset problems which at a first glance do not look as if they could be captured by this definition. An example is the Cycle Packing problem. Here the input is a graph G and integer k, and the task is to determine whether G contains k pairwise vertex-disjoint cycles $C_1, C_2, \ldots, C_k$. This is a vertex subset maximization problem because G has k vertex-disjoint cycles if and only if there exists a set $S \subseteq V(G)$ of size at least k and $\phi(G, S)$ is true, where $\phi(G, S)$ is defined such that $\phi(G, S) = \mathsf{true} \iff G$ contains a subgraph G' such that each connected component of G' is a cycle and each connected component of G' contains exactly one vertex from S.

The above definition of Cycle Packing may seem bizarre, since checking whether $\phi(G, S)$ is true for a given graph G and set S is NP-complete. In fact this problem is considered as a more difficult problem than Cycle Packing. Nevertheless, this definition shows that Cycle Packing is indeed a subset problem.

3.2 Optimality functions

For any vertex or edge subset minimization problem Π we have that $(G, k) \in \Pi$ implies that $(G, k') \in \Pi$ for all $k' \geq k$. Similarly, for a vertex or edge subset maximization problem we have that $(G, k) \in \Pi$ implies that $(G, k') \in \Pi$ for all $k' \leq k$. Thus the notion of "optimality" is well defined for subset problems.

▶ **Definition 2.** For a vertex or edge subset minimization problem Π, we define

$$OPT_\Pi(G) = \min \{k \ : \ (G, k) \in \Pi\}.$$

If no k such that $(G, k) \in \Pi$ exists, $OPT_\Pi(G)$ returns $+\infty$. For a vertex or edge subset maximization problem Π,

$$OPT_\Pi(G) = \max \{k \ : \ (G, k) \in \Pi\}.$$

If no k such that $(G, k) \in \Pi$ exists, $OPT_\Pi(G)$ returns $-\infty$. We define $SOL_\Pi(G)$ to be a function that, given as an input a graph G, returns a set S of size $OPT_\Pi(G)$ such that $\phi(G, S) = \mathsf{true}$, and returns **null** if no such set S exists.

▶ **Definition 3.** A subset problem Π is *contraction-closed* (resp. *minor-closed*) if for any two graphs G_1 and G_2 it holds that $G_1 \leq_c G_2 \Rightarrow OPT_\Pi(G_1) \leq OPT_\Pi(G_1)$ (resp. $G_1 \leq_m G_2 \Rightarrow OPT_\Pi(G_1) \leq OPT_\Pi(G_2)$).

3.3 Bidimensional problems

We are now ready to introduce the concept of bidimensionality.

▶ **Definition 4** (Bidimensional problem)**.** A subset problem Π is

minor-bidimensional if
- Π is minor-closed, and
- $\lim_{k\to\infty} \frac{OPT_\Pi(\boxplus_k)}{k^2} = \delta > 0$

contraction-bidimensional if
- Π is contraction-closed, and
- $\lim_{k\to\infty} \frac{OPT_\Pi(\Gamma_k)}{k^2} = \delta > 0$

In each of the above cases (when applicable), we say that the positive real δ is the *density* of the problem Π. A subset problem Π is *bidimensional* if it is minor or contraction bidimensional.

Examples of bidimensional subset problems are (Connected) Vertex Cover, (Connected) Feedback Vertex Set, Induced Matching, Longest Cycle, (Induced) Cycle Packing, d-Scattered Set, Longest Path, (Connected) r-Dominating Set, Diamond Hitting Set, Face Cover, (Connected) Edge Dominating Set, and Unweighted TSP Tour.

It is usually quite easy to determine whether a problem is contraction (or minor) bidimensional. Take as an example Independent Set. Contracting an edge may never increase the size of the maximum independent set, so the problem is contraction-closed. Furthermore it is easy to verify that Γ_k contains an independent set of size $\frac{(k-1)^2}{4}$. Thus Independent Set is contraction-bidimensional with density $1/4$. On the other hand deleting edges may increase the size of a maximum-size independent set in G. Thus Independent Set is not minor-bidimensional.

4 Subexponential parameterized algorithms

A central problem in parameterized algorithm design is to investigate in which cases and under which input restrictions a parameterized problem belongs to FPT and, if so, to find algorithms with the simplest possible parameter dependence.

Let Π be a parameterized graph problem in FPT that can be solved in $f(k) \cdot n^{O(1)}$ steps[1]. When $f(k) = 2^{o(k)}$ we say that Π admits a *subexponential parameterized algorithm* (see [44] for a survey on subexponential parameterized algorithms).

In [14], Cai and Juedes proved that several parameterized problems do not admit subexponential parameterized algorithms, unless 3-SAT can be solved in time subexponential in the number of its variables[2]. Among them, one can distinguish core problems such as the standard parameterizations of Vertex Cover, Dominating Set, and Feedback Vertex Set. However, it appears that many parameterized problems admit subexponential parameterized algorithms running in $2^{O(\sqrt{k})} \cdot n^{O(1)}$ steps when their inputs are restricted to planar graphs or other classes of surface embeddable graphs. Moreover, the results of [14] indicated that the parameterized dependence $2^{O(\sqrt{k})}$ is the best we may expect when the planarity restriction is imposed. The first subexponential parameterized algorithm on planar graphs appeared in [1] for Dominating Set, Independent Dominating Set, and Face Cover. After that, subexponential parameterized algorithms where designed for many other problems [85, 31, 1, 71, 19, 29, 48, 49, 70, 62, 72, 37, 24, 58]. Most of these results are now covered by the main result of this section (Theorem 7).

4.1 Singly exponentially solvable problems w.r.t. treewidth

Let Π be a subset problem Π. We say that Π is *singly exponentially solvable with respect to treewidth* if there exists an algorithm that computes $OPT_\Pi(G)$ in $2^{O(\mathbf{tw}(G))}n^{O(1)}$ steps.

Typically, to prove that a subset problem Π is singly exponentially solvable with respect to treewidth requires the design of dynamic programming algorithms on tree decompositions of width at most w whose tables are of singly exponential size on w. The design of such algorithms has occupied a lot of research in parameterized complexity [8, 87, 83, 13, 3, 5, 40, 6, 25, 45, 42, 43, 84]. In most of the cases, such algorithms run in $2^{O(\mathbf{tw}(G))}n$ steps. A general meta-algorithmic condition implying that a problem is singly exponentially solvable with respect to treewidth was given in [78] and is a model of Modal Logic called *Existential Counting Modal Logic* (ECM-Logic).

4.2 Bidimensionality and subexponential parameterized algorithms

Let Π be a vertex/edge subset minimization (resp. maximization) problem. Consider the following two conditions for Π.

A [*Algorithmic*] Π is singly exponentially solvable with respect to treewidth.

B [*Combinatorial*] If (G, k) is a YES- (resp. NO-) instance of Π, then $\mathbf{tw}(G) = o(k)$.

▶ **Proposition 5.** *If Π is a vertex/edge subset minimization (resp. maximization) problem satisfying conditions **A** and **B**, then Π admits a subexponential parameterized algorithm.*

Proof. Let (G, k) be an input for Π. If the treewidth of the input graph exceeds the upper bound of the combinatorial condition **B**, then we can safely report that (G, k) is a NO- (resp.

[1] From now on, we use n to denote the number of vertices of the input graph G, i.e., $n = |V(G)|$.

[2] This is hypothesis is also known as the Exponential Time Hypothesis (ETH).

YES-) instance and we are done. This step can be supported by the algorithm in [9] that, given a graph G and an integer w, either returns that $\mathbf{tw}(G) > w$, or outputs a tree-decomposition of G of width $\leq 5w$. If now the bound of the combinatorial condition **B** holds, we have a tree-decomposition of G of width $5 \cdot \mathbf{tw}(G) = o(k)$ and the result follows directly from the algorithmic condition **A**. ◀

The following is an important combinatorial consequence of bidimensionality. It reflects the original idea of [27].

▶ **Proposition 6.** *If Π is a subset problem that is minor (resp. contraction) bidimensional and $\mathcal{G}$ is a graph class with the **SQGM** (resp. **SQGC**) property, then $\Pi \Cap \mathcal{G}$ satisfies the combinatorial condition **B**.*

Proof. We give the proof in the case where Π is a vertex/edge subset minimization problem. For this, we have to show that if (G, k) is a YES-instance of $\Pi \Cap \mathcal{G}$, then $\mathbf{tw}(G) = o(k)$. Indeed, if $(G, k) \in \Pi$ then

$$OPT_{\Pi}(G) \leq k. \tag{1}$$

$$\text{If } \boxplus_r \leq_m G, \text{ then } \quad OPT_{\Pi}(\boxplus_r) \leq OPT_{\Pi}(G). \tag{2}$$

As Π is minor (resp. contraction) bidimensional, then

$$OPT_{\Pi}(\boxplus_r) = \Omega(r^2). \tag{3}$$

From (1), (2), and (3), it follows that if $\boxplus_r \leq_m G$, then $r = O(\sqrt{k})$ which, from the **SQGM** (resp. **SQGC**) property of $\mathcal{G}$ implies that $\mathbf{tw}(G) = o(k)$. ◀

Using Propositions 5 and 6, we easily conclude with the following.

▶ **Theorem 7.** *Let Π be a vertex/edge subset minimization (resp. maximization) problem that*

i. *is singly exponentially solvable with respect to treewidth and*
ii. *is minor- (resp. contraction-) bidimensional*

*and let $\mathcal{G}$ be a graph class with the **SQGM** (resp. **SQGC**) property. Then the restriction of Π to $\mathcal{G}$ admits a* subexponential parameterized algorithm.

Notice that the above theorem can become purely meta-algorithmic if we replace condition *i.* by the expressibility of Π in ECM-Logic, as indicated by the results in [78]. Clearly, for the applicability of the above approach, it is important to detect graph classes with the **SQGM** (resp. **SQGC**) property. Historically, this was first done for bounded genus graphs in [27] and in [38], for H-minor free graphs in [35], [28], and [54], and for families of geometric graphs in [57] and [67]. Finally, results that either use ideas similar to bidimensionality o provide alternative techniques for the derivation of subexponential (or low-exponential) parameterized algorithms have been examined in [47, 56, 79, 41, 80, 66].

5 Kernelization

Kernelization has been extensively studied in parameterized complexity. It can be seen as the strategy of analyzing preprocessing or data reduction routines from a parameterized complexity perspective.

5.1 Kernelization algorithms

The notion of *kernelization* is formally defined as follows.

▶ **Definition 8.** A *kernelization algorithm*, or simply a *kernel*, for a parameterized problem Π is an algorithm $\mathcal{A}$ that, given an instance (x, k) of Π, runs in polynomial, on $|x|$, time and outputs an equivalent instance (x', k') of Π where $|x'| + k' \leq g(k)$ for some computable function $g : \mathbb{N} \to \mathbb{N}$ called the *size* of the kernel. In this case we say that Π admits a g *kernel* and if the size g is a polynomial (resp. linear) function of the parameter k, then we say that Π admits a *polynomial (resp. linear) kernel*. As we agreed for parameterized graph problems, we will assume that x corresponds to a graph and we treat the size of a kernel as a function on the number of vertices of the graph in the equivalent instance.

Notable examples of known kernels are a $2k$ kernel for Vertex Cover [18], a $355k$ kernel for Dominating Set on planar graphs [2], which later was improved to a $67k$ kernel [17] and an $O(k^2)$ kernel for Feedback Vertex Set [86] parameterized by the solution size. One of the most intensively studied directions in kernelization is the study of problems on planar graphs and other classes of sparse graphs. This study was initiated by Alber et al. [2] who gave the first linear-sized kernel for the Dominating Set problem on planar graphs. The work of Alber et al. [2] triggered an explosion of papers on kernelization, and kernels of linear sizes were obtained for a variety of parameterized problems on planar graphs including Connected Vertex Cover, Minimum Edge Dominating Set, Maximum Triangle Packing, Efficient Edge Dominating Set, Induced Matching, Full-Degree Spanning Tree, Feedback Vertex Set, Cycle Packing, Blue-Red Dominating Set, and Connected Dominating Set [2, 11, 12, 17, 51, 68, 69, 75, 76, 64, 60]. Most of these results are now covered by the main result of this section (Theorem 13). We refer to the surveys [73, 74] for a detailed exposition of the area of kernelization.

5.2 Separability

We now restrict our attention to problems Π that are somewhat well-behaved in the sense that whenever we have a small separator in the graph that splits the graph in two parts L and R, the intersection $|X \cap L|$ of L with any optimal solution X to the entire graph is a good estimate of $OPT_\Pi(G[L])$. This behavior is called *separability* and variants of it have been used, combined with bidimensionality, for the derivation of Efficient Polynomial Time Approximation Schemes (EPTAS), see [33, 55].

▶ **Definition 9** (Separability)**.** Let $f : \mathbb{N} \to \mathbb{N}$. We say that a subset problem Π is *f-separable* if for any graph G and $L \subseteq V(G)$ such that $|\partial_G(L)| \leq t$, it holds that

$$|SOL_\Pi(G) \cap L| - f(t) \leq OPT_\Pi(G[L]) \leq |SOL_\Pi(G) \cap L| + f(t).$$

Π is called *separable* if there exists a function f such that Π is *f-separable*. Π is called *linearly separable* if it is *f-separable* for some linear function f.

5.3 Protrusion decompositions and replacements

We introduce the notions of *protrusion*, *protrusion decomposition*, and *protrusion replacement*.

Protrusion decompositions. Given a graph G, we say that a set $X \subseteq V(G)$ is an t-protrusion of G if $|\partial(X)| \leq t$ and $\mathbf{tw}(G[X]) \leq t$. An (α, β)*-protrusion decomposition* of a graph G is a partition $\mathcal{P} = \{R_0, R_1, \ldots, R_\rho\}$ of $V(G)$ such that

- $\max\{\rho, |R_0|\} \leq \alpha$,
- each $R_i^+ = N_G[R_i]$, $i \in \{1, \ldots, \rho\}$, is a β-protrusion of G, and
- for every $i \in \{1, \ldots, \rho\}$, $N_G(R_i) \subseteq R_0$.

Protrusion replacement algorithms. Let Π be a parameterized graph problem and let $f : \mathbb{Z}^+ \to \mathbb{Z}^+$ be a non-decreasing function. An *f-protrusion replacement family* for Π is a collection $\mathcal{A} = \{\mathsf{A}_i \mid i \geq 0\}$ of algorithms, such that algorithm A_i receives as input a pair (I, X), where $I = (G, k)$ is an instance of Π and X is an i-protrusion of G with at least $f(i)$ vertices and outputs an equivalent instance $I^* = (G^*, k^*)$ where $|V(G^*)| < |V(G)|$ and $k^* \leq k$. We say that Π has a *protrusion replacement family* if it has a f-protrusion replacement family for some $f : \mathbb{Z}^+ \to \mathbb{Z}^+$.

5.4 Meta-algorithmic results for kernels

Let Π be a vertex/edge subset minimization (resp. maximization) problem. The following two conditions for such a problem Π were defined in [10, 7]. They can be seen as the "kernelization counterparts" of the properties **A** and **B** that we introduced in Subsection 4.2.

A [*Algorithmic*] Π has a protrusion replacement family.

B [*Combinatorial*] If (G, k) is a YES- (resp. NO-) instance of Π, then G has an $(O(k), O(1))$-protrusion decomposition.

The next result is a special case of Theorem 4.6 in [10, 7].

▶ **Proposition 10.** *If a parameterized graph problem Π has properties **A** and **B**, then Π admits a linear kernel.*

The following result is based on the property that a problem has Finite Integer Index (FII). In [10, Lemma 5.19] it was proved that this problem property is able to yield property **A** and, as it has recently been proved in [61], FII is a consequence of CMSO expressibility and the separability property.

▶ **Proposition 11.** *Every* MIN/MAX-CMSO *subset problem Π that is linearly separable has property **A**.*

We now present one of the main combinatorial consequences of bidimensionality. It has been proved in [59, 61].

▶ **Proposition 12.** *Let $\mathcal{G}$ be a graph class with the **SQGM** (resp. **SQGC**) property and let Π be a subset problem that is minor- (resp. contraction-) bidimensional and linear-separable. Then $\Pi \Cap \mathcal{G}$ satisfies property **B**.*

Using Propositions 10, 11, and 12, one can easily derive the following meta-algorithmic result.

▶ **Theorem 13.** *Let Π be a subset problem that*

i. *is a* MIN/MAX-CMSO *problem,*

ii. *is minor- (resp. contraction-) bidimensional,*

iii. *is linearly separable,*

*and let $\mathcal{G}$ is a graph class with the **SQGM** (resp. **SQGC**) property. Then the restriction of Π to $\mathcal{G}$ admits a linear kernel.*

6 Further extensions

In this paper we presented two consequences of bidimensionality, namely Theorem 7 (subexponential parameterized algorithms) and 13 (linear kernelization). Further applications of bidimensionality on the automatic derivation of EPTAS can be found in [33] and [55]. It is an interesting question whether this problem property can be exploited to other algorithmic paradigms.

For the existing applications, there are two main directions. The first is to enlarge the set of graph classes satisfying the **SQGM** or the **SQGC** property. A first step, escaping from the graph minors framework, are the results of [57] and [67] on geometric graphs. Another direction is to make the constants involved in Theorems 7 and 13 explicit so as to optimize the running times of the derived algorithms. A step is this direction was taken in [63] using dynamic programming for certain families of problems. It is also interesting to build extensions of bidimensionality for problems that instead of being closed under minors or contractions are closed under some other partial ordering on graphs such as topological minors, immersions, induced minors and others. We believe that recent results such as those in [65, 88, 16] might be helpful starting points in this direction.

Acknowledgments. I am thankful to Fedor V. Fomin, Stavros G. Kolliopoulos, and Spyros Maniatis, for their helpful remarks on the manuscript.

References

1 J. Alber, H. L. Bodlaender, H. Fernau, T. Kloks, and R. Niedermeier. Fixed parameter algorithms for dominating set and related problems on planar graphs. *Algorithmica*, 33(4):461–493, 2002.

2 J. Alber, M. R. Fellows, and R. Niedermeier. Polynomial-time data reduction for dominating sets. *J. ACM*, 51:363–384, 2004.

3 J. Alber and R. Niedermeier. Improved tree decomposition based algorithms for domination-like problems. In S. Rajsbaum, editor, *LATIN 2002: Theoretical Informatics*, volume 2286 of *LNCS*, pages 221–233. Springer Berlin / Heidelberg, 2002.

4 S. Arnborg, J. Lagergren, and D. Seese. Easy problems for tree-decomposable graphs. *Journal of Algorithms*, 12:308–340, 1991.

5 N. Betzler, R. Niedermeier, and J. Uhlmann. Tree decompositions of graphs: Saving memory in dynamic programming. *Discrete Optimization*, 3(3):220 – 229, 2006. Graphs and Combinatorial Optimization.

6 A. Björklund, T. Husfeldt, P. Kaski, and M. Koivisto. Fourier meets Möbius: fast subset convolution. In *STOC*, pages 67–74. ACM, 2007.

7 H. Bodlaender, F. Fomin, D. Lokshtanov, E. Penninkx, S. Saurabh, and D. Thilikos. (Meta) kernelization. In *50th Annual IEEE Symposium on Foundations of Computer Science, (FOCS 2009)*. ACM, 2009.

8 H. L. Bodlaender. Dynamic programming on graphs with bounded treewidth. In *15th International Colloquium on Automata, Languages and Programming (ICALP)*, volume 317 of *LNCS*, pages 105–118. Springer-Verlag, 1988.

9 H. L. Bodlaender, P. G. Drange, M. S. Dregi, F. V. Fomin, D. Lokshtanov, and M. Pilipczuk. An $O(c^k n)$ 5-approximation algorithm for treewidth. In *54th Annual IEEE Symposium on Foundations of Computer Science, FOCS 2013, 26-29 October, 2013, Berkeley, CA, USA*, pages 499–508, 2013.

10 H. L. Bodlaender, F. V. Fomin, D. Lokshtanov, E. Penninkx, S. Saurabh, and D. M. Thilikos. (meta) kernelization. *CoRR*, abs/0904.0727, 2009.

11 H. L. Bodlaender and E. Penninkx. A linear kernel for planar feedback vertex set. In *Proccedings of the 3rd International Workshop on Exact and Parameterized Computation (IWPEC 2008)*, volume 5018 of *LNCS*, pages 160–171. Springer, Berlin, 2008.

12 H. L. Bodlaender, E. Penninkx, and R. B. Tan. A linear kernel for the k-disjoint cycle problem on planar graphs. In *19th International Symposium on Algorithms and Computation (ISAAC)*, volume 5369 of *LNCS*, pages 306–317. Springer, 2008.

13 H. L. Bodlaender and J. A. Telle. Space-efficient construction variants of dynamic programming. *Nordic J. Comput.*, 11(4):374–385, 2004.

14 L. Cai and D. Juedes. On the existence of subexponential parameterized algorithms. *J. Comput. System Sci.*, 67(4):789 – 807, 2003.

15 C. Chekuri and J. Chuzhoy. Polynomial bounds for the grid-minor theorem. *CoRR*, abs/1305.6577, 2013.

16 C. Chekuri and J. Chuzhoy. Degree-3 treewidth sparsifiers. In *Proceedings of the Twenty-Sixth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2015, San Diego, CA, USA, January 4-6, 2015*, pages 242–255, 2015.

17 J. Chen, H. Fernau, I. A. Kanj, and G. Xia. Parametric duality and kernelization: Lower bounds and upper bounds on kernel size. *SIAM Journal on Computing*, 37:1077–1106, 2007.

18 J. Chen, I. A. Kanj, and W. Jia. Vertex cover: further observations and further improvements. *J. Algorithms*, 41(2):280–301, 2001.

19 J. Chen, I. A. Kanj, L. Perković, E. Sedgwick, and G. Xia. Genus characterizes the complexity of certain graph problems: Some tight results. *Journal of Computer and System Sciences*, 73(6):892 – 907, 2007.

20 J. Chuzhoy. Excluded grid theorem: Improved and simplified. In *Proceedings of the Forty-Seventh Annual ACM on Symposium on Theory of Computing*, STOC '15, pages 645–654, New York, NY, USA, 2015. ACM.

21 B. Courcelle. The monadic second-order logic of graphs. I. Recognizable sets of finite graphs. *Information and Computation*, 85(1):12–75, 1990.

22 B. Courcelle. The expression of graph properties and graph transformations in monadic second-order logic. *Handbook of Graph Grammars*, pages 313–400, 1997.

23 M. Cygan, F. V. Fomin, L. Kowalik, D. Lokshtanov, D. Marx, M. Pilipczuk, M. Pilipczuk, and S. Saurabh. *Parameterized Algorithms*. Springer, 2015.

24 M. Cygan, S. Kratsch, and J. Nederlof. Fast hamiltonicity checking via bases of perfect matchings. In *Proceedings of the Forty-fifth Annual ACM Symposium on Theory of Computing*, STOC '13, pages 301–310, New York, NY, USA, 2013. ACM.

25 M. Cygan, J. Nederlof, M. Pilipczuk, M. Pilipczuk, J. M. M. van Rooij, and J. O. Wojtaszczyk. Solving connectivity problems parameterized by treewidth in single exponential time. In *IEEE 52nd Annual Symposium on Foundations of Computer Science, (FOCS 2011)*, pages 150–159. IEEE Computer Society, 2011.

26 E. D. Demaine. Algorithmic graph minors and bidimensionality. In *Graph Theoretic Concepts in Computer Science - 36th International Workshop, WG 2010, Zarós, Crete, Greece, June 28-30, 2010 Revised Papers*, page 2, 2010.

27 E. D. Demaine, F. V. Fomin, M. Hajiaghayi, and D. M. Thilikos. Subexponential parameterized algorithms on graphs of bounded-genus and H-minor-free graphs. In *Fifteenth Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 830–839 (electronic). ACM, New York, 2004.

28 E. D. Demaine, F. V. Fomin, M. Hajiaghayi, and D. M. Thilikos. Bidimensional parameters and local treewidth. *SIAM J. Discrete Math.*, 18(3):501–511, 2005.

29 E. D. Demaine, F. V. Fomin, M. Hajiaghayi, and D. M. Thilikos. Fixed-parameter algorithms for (k, r)-center in planar graphs and map graphs. *ACM Trans. Algorithms*, 1(1):33–47, 2005.

30 E. D. Demaine, F. V. Fomin, M. Hajiaghayi, and D. M. Thilikos. Bidimensional structures: Algorithms, combinatorics and logic (dagstuhl seminar 13121). *Dagstuhl Reports*, 3(3):51–74, 2013.

31 E. D. Demaine, F. V. Fomin, M. T. Hajiaghayi, and D. M. Thilikos. Fixed-parameter algorithms for (k, r)-center in planar graphs and map graphs. *ACM Transactions on Algorithms*, 1(1):33–47, 2005.

32 E. D. Demaine and M. Hajiaghayi. Bidimensionality, map graphs, and grid minors. *CoRR*, abs/cs/0502070, 2005.

33 E. D. Demaine and M. Hajiaghayi. Bidimensionality: new connections between fpt algorithms and ptass. In *16th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA 2005)*, pages 590–601. ACM-SIAM, New York, 2005.

34 E. D. Demaine and M. Hajiaghayi. The bidimensionality theory and its algorithmic applications. *Comput. J.*, 51(3):292–302, 2008.

35 E. D. Demaine and M. Hajiaghayi. Linearity of grid minors in treewidth with applications through bidimensionality. *Combinatorica*, 28(1):19–36, 2008.

36 E. D. Demaine, M. Hajiaghayi, and K. Kawarabayashi. Algorithmic graph minor theory: Improved grid minor bounds and Wagner's contraction. *Algorithmica*, 54(2):142–180, 2009.

37 E. D. Demaine, M. Hajiaghayi, and D. M. Thilikos. Exponential speedup of fixed-parameter algorithms for classes of graphs excluding single-crossing graphs as minors. *Algorithmica*, 41:245–267, 2005.

38 E. D. Demaine, M. Hajiaghayi, and D. M. Thilikos. The Bidimensional Theory of Bounded-Genus Graphs. *SIAM Journal on Discrete Mathematics*, 20(2):357–371, 2006.

39 E. D. Demaine and M. T. Hajiaghayi. Fast algorithms for hard graph problems: Bidimensionality, minors, and local treewidth. In *Graph Drawing, 12th International Symposium, GD 2004, New York, NY, USA, September 29 - October 2, 2004, Revised Selected Papers*, pages 517–533, 2004.

40 F. Dorn. Dynamic programming and fast matrix multiplication. In *14th Annual European Symposium on Algorithms (ESA 2006)*, volume 4168 of *LNCS*, pages 280–291. Springer, Berlin, 2006.

41 F. Dorn, F. V. Fomin, D. Lokshtanov, V. Raman, and S. Saurabh. Beyond bidimensionality: Parameterized subexponential algorithms on directed graphs. *Information and Computation*, 233(0):60 – 70, 2013.

42 F. Dorn, F. V. Fomin, and D. M. Thilikos. Fast subexponential algorithm for non-local problems on graphs of bounded genus. In *10th Scandinavian Workshop on Algorithm Theory (SWAT 2006)*, volume 4059 of *LNCS*, pages 172–183. Springer, Berlin, 2006.

43 F. Dorn, F. V. Fomin, and D. M. Thilikos. Catalan structures and dynamic programming in H-minor-free graphs. In *ACM-SIAM Symposium on Discrete Algorithms (SODA 2008)*, pages 631–640. SIAM, 2008.

44 F. Dorn, F. V. Fomin, and D. M. Thilikos. Subexponential parameterized algorithms. *Computer Science Review*, 2(1):29–39, 2008.

45 F. Dorn, E. Penninkx, H. L. Bodlaender, and F. V. Fomin. Efficient exact algorithms on planar graphs: Exploiting sphere cut decompositions. *Algorithmica*, 58(3):790–810, 2010.

46 R. G. Downey and M. R. Fellows. *Parameterized complexity.* Monographs in Computer Science. Springer-Verlag, New York, 1999.

47 F. F. Dragan, F. V. Fomin, and P. A. Golovach. Spanners in sparse graphs. In *35th International Colloquium on Automata, Languages and Programming (ICALP 2008)*, volume 5125 of *LNCS*, pages 597–608. Springer, 2008.

48 H. Fernau. Graph separator algorithms: a refined analysis. In *Graph-theoretic Concepts in Computer Science*, volume 2573 of *LNCS*, pages 186–197. Springer, Berlin, 2002.

49 H. Fernau and D. Juedes. A geometric approach to parameterized algorithms for domination problems on planar graphs. In *29th International Symposium on Mathematical Foundations of Computer (MFCS 2004)*, volume 3153 of *LNCS*, pages 488–499. Springer, Berlin, 2004.

50 J. Flum and M. Grohe. *Parameterized Complexity theory.* Texts in Theoretical Computer Science. An EATCS Series. Springer-Verlag, Berlin, 2006.

51 F. V. Fomin, S. S. Daniel Lokshtanov, and D. M. Thilikos. Linear kernels for (connected) dominating set on H-minor-free graphs. In *23st ACM–SIAM Symposium on Discrete Algorithms (SODA 2012)*. ACM-SIAM, San Francisco, California, 2012.

52 F. V. Fomin, E. D. Demaine, and M. T. Hajiaghayi. Bidimensionality. In *Encyclopedia of Algorithms*. Springer, 2015.

53 F. V. Fomin, P. Golovach, and D. M. Thilikos. Contraction bidimensionality: the accurate picture. In *17th Annual European Symposium on Algorithms (ESA 2009)*, LNCS, pages 706–717. Springer, 2009.

54 F. V. Fomin, P. A. Golovach, and D. M. Thilikos. Contraction obstructions for treewidth. *J. Comb. Theory, Ser. B*, 101(5):302–314, 2011.

55 F. V. Fomin, D. Lokshtanov, V. Raman, and S. Saurabh. Bidimensionality and EPTAS. *CoRR*, abs/1005.5449, 2010.

56 F. V. Fomin, D. Lokshtanov, V. Raman, and S. Saurabh. Subexponential algorithms for partial cover problems. *Inf. Process. Lett.*, 111(16):814–818, 2011.

57 F. V. Fomin, D. Lokshtanov, and S. Saurabh. Bidimensionality and geometric graphs. In *Proceedings of the Twenty-Third Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2012, Kyoto, Japan, January 17-19, 2012*, pages 1563–1575, 2012.

58 F. V. Fomin, D. Lokshtanov, and S. Saurabh. Efficient computation of representative sets with applications in parameterized and exact algorithms. In *Proceedings of the Twenty-Fifth Annual ACM-SIAM Symposium on Discrete Algorithms*, SODA '14, pages 142–151. SIAM, 2014.

59 F. V. Fomin, D. Lokshtanov, S. Saurabh, and D. M. Thilikos. Bidimensionality and kernels. In *21st Annual ACM-SIAM Symposium on Discrete Algorithms (SODA 2010)*, pages 503–510. ACM-SIAM, 2010.

60 F. V. Fomin, D. Lokshtanov, S. Saurabh, and D. M. Thilikos. Linear kernels for (connected) dominating set on graphs with excluded topological subgraphs. In N. Portier and T. Wilke, editors, *30th International Symposium on Theoretical Aspects of Computer Science (STACS 2013)*, volume 20 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 92–103, Dagstuhl, Germany, 2013. Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik.

61 F. V. Fomin, D. Lokshtanov, S. Saurabh, and D. M. Thilikos. Bidimensionality and kernels. Revised manuscript, 2015.

62 F. V. Fomin and D. M. Thilikos. Dominating sets in planar graphs: branch-width and exponential speed-up. *SIAM J. Comput.*, 36(2):281–309 (electronic), 2006.

63 V. Garnero, C. Paul, I. Sau, and D. M. Thilikos. Explicit linear kernels via dynamic programming. In *31st International Symposium on Theoretical Aspects of Computer Science (STACS 2014), STACS 2014, March 5-8, 2014, Lyon, France*, pages 312–324, 2014.

64 V. Garnero, I. Sau, and D. M. Thilikos. A linear kernel for planar red-blue dominating set. In *Proceedings of the 12th Cologne Twente Workshop on Graphs and Combinatorial Optimization (CTW)*, pages 117–120, Enschede, The Netherlands, May 2013.

65 A. C. Giannopoulou and D. M. Thilikos. Optimizing the graph minors weak structure theorem. *SIAM J. Discrete Math.*, 27(3):1209–1227, 2013.

66 P. A. Golovach, M. Kamiński, D. Paulusma, and D. M. Thilikos. Induced packing of odd cycles in a planar graph. In *20th International Symposium on Algorithms and Computation (ISAAC 2009)*, volume 5878 of *LNCS*, pages 514–523. Springer, Berlin, 2009.

67 A. Grigoriev, A. Koutsonas, and D. Thilikos. Bidimensionality of geometric intersection graphs. In V. Geffert, B. Preneel, B. Rovan, J. Štuller, and A. Tjoa, editors, *SOFSEM 2014: Theory and Practice of Computer Science*, volume 8327 of *Lecture Notes in Computer Science*, pages 293–305. Springer International Publishing, 2014.

68 J. Guo, R. Niedermeier, and S. Wernicke. Fixed-parameter tractability results for full-degree spanning tree and its dual. In *Second International Workshop on Parameterized and Exact Computation (IWPEC)*, volume 4169 of *LNCS*, pages 203–214. Springer, 2006.

69 I. Kanj, M. J. Pelsmajer, M. Schaefer, and G. Xia. On the induced matching problem. *Journal of Computer and System Sciences*, 77(6):1058 – 1070, 2011.

70 I. Kanj and L. Perković. Improved parameterized algorithms for planar dominating set. In *27th International Symposium Mathematical Foundations of Computer Science (MFCS 2002)*, volume 2420 of *LNCS*, pages 399–410. Springer, Berlin, 2002.

71 T. Kloks, C. M. Lee, and J. Liu. New algorithms for k-face cover, k-feedback vertex set, and k-disjoint cycles on plane and planar graphs. In *28th International Workshop on Graph Theoretic Concepts in Computer Science (WG 2002)*, volume 2573 of *LNCS*, pages 282–295. Springer, Berlin, 2002.

72 A. Koutsonas and D. M. Thilikos. Planar feedback vertex set and face cover: Combinatorial bounds and subexponential algorithms. *Algorithmica*, 60(4):987–1003, 2011.

73 S. Kratsch. Recent developments in kernelization: A survey. *Bulletin of the EATCS*, 113, 2014.

74 D. Lokshtanov, N. Misra, and S. Saurabh. Kernelization - preprocessing with a guarantee. In *The Multivariate Algorithmic Revolution and Beyond - Essays Dedicated to Michael R. Fellows on the Occasion of His 60th Birthday*, pages 129–161, 2012.

75 D. Lokshtanov, M. Mnich, and S. Saurabh. Linear kernel for planar connected dominating set. In *6th Annual Conference on Theory and Applications of Models of Computation (TAMC 2009)*, volume 5532 of *LNCS*, pages 281–290. Springer, 2009.

76 H. Moser and S. Sikdar. The parameterized complexity of the induced matching problem in planar graphs. In *Proceedings First Annual International WorkshopFrontiers in Algorithmics (FAW)*, volume 4613 of *LNCS*, pages 325–336. Springer, 2007.

77 R. Niedermeier. *Invitation to fixed-parameter algorithms*, volume 31 of *Oxford Lecture Series in Mathematics and its Applications*. Oxford University Press, Oxford, 2006.

78 M. Pilipczuk. Problems parameterized by treewidth tractable in single exponential time: A logical approach. In F. Murlak and P. Sankowski, editors, *Mathematical Foundations of Computer Science 2011*, volume 6907 of *Lecture Notes in Computer Science*, pages 520–531. Springer Berlin Heidelberg, 2011.

79 M. Pilipczuk, M. Pilipczuk, P. Sankowski, and E. J. van Leeuwen. Subexponential-time parameterized algorithm for steiner tree on planar graphs. In *30th International Symposium on Theoretical Aspects of Computer Science, STACS 2013, February 27 - March 2, 2013, Kiel, Germany*, pages 353–364, 2013.

80 M. Pilipczuk, M. Pilipczuk, P. Sankowski, and E. J. van Leeuwen. Network sparsification for steiner problems on planar and bounded-genus graphs. In *55th IEEE Annual Symposium on Foundations of Computer Science, FOCS 2014, Philadelphia, PA, USA, October 18-21, 2014*, pages 276–285, 2014.

81 N. Robertson and P. D. Seymour. Graph Minors. V. Excluding a planar graph. *Journal of Combinatorial Theory, Series B*, 41(2):92–114, 1986.

82 N. Robertson, P. D. Seymour, and R. Thomas. Quickly excluding a planar graph. *J. Combin. Theory Ser. B*, 62(2):323–348, 1994.

83 J. Rué, I. Sau, and D. M. Thilikos. Dynamic programming for graphs on surfaces. In *Automata, Languages and Programming, 37th International Colloquium, ICALP 2010 (1)*, volume 6198 of *Lecture Notes in Computer Science*, pages 372–383. Springer, 2010.

84 J. Rué, I. Sau, and D. M. Thilikos. Asymptotic enumeration of non-crossing partitions on surfaces. *Discrete Mathematics*, 313(5):635–649, 2013.

85 D. M. Thilikos. Fast sub-exponential algorithms and compactness in planar graphs. In *19th Annual European Symposium on Algorithms (ESA 2011)*, pages 358–369, 2011.

86 S. Thomassé. A $4 \cdot k^2$ kernel for feedback vertex set. *ACM Trans. Algorithms*, 6(2):32:1–32:8, Apr. 2010.

87 J. M. M. van Rooij, H. L. Bodlaender, and P. Rossmanith. Dynamic programming on tree decompositions using generalised fast subset convolution. In *ESA*, pages 566–577. Springer-Verlag, 2009.

88 P. Wollan. The structure of graphs not admitting a fixed immersion. *J. Comb. Theory, Ser. B*, 110:47–66, 2015.